



PYSCRIPT: O PYTHON NO SEU BROWSER

Já pensou executar código Python no seu Browser?! Com PyScript, agora isso é possível! Venha ver como neste Post!

Crie Ebooks técnicos incríveis em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs



Deixe que nossa IA faça o trabalho pesado

 Syntax Highlight

 Adicione Banners Promocionais

 Edite em Markdown em Tempo Real

 Infográficos feitos por IA

TESTE AGORA 

Salve salve queridos Pythonistas!

Já pensou em poder executar código Python **diretamente** no browser?! 😱

Pois é, com o PyScript **agora isto é possível!**

o PyScript foi introduzido na PyCon US 2022 e vem chamando bastante a atenção dos amantes do Python!

E aí, será que essa moda de executar código Python diretamente no seu navegador vai pegar?

Mas primeiro, vamos aprender como podemos escrever código Python dentro de código HTML!

Faça seu cafezinho e bora nessa!

De onde surgiu essa ideia maluca...

Durante uma palestra na PyCon US 2022, o CEO da Anaconda, Peter Wang, revelou um projeto bastante surpreendente: o **PyScript**.

O PyScript é uma biblioteca JavaScript que permite aos usuários criar aplicativos Python no navegador usando uma mistura de Python e HTML padrão.

O objetivo final do projeto é permitir que um público muito mais amplo (por exemplo, desenvolvedores front-end) se beneficie do poder do Python e de suas várias bibliotecas (estatística, ML/DL, etc.).

É possível usar alguns dos componentes de UI pré-definidos, como por exemplo: botões, contêineres, caixas de texto etc.

Além disso, não é necessário nos preocupar com a implantação, pois com o PyScript tudo acontecerá em navegadores da web.

Se você é Cientista de Dados, podem - por exemplo - compartilhar arquivos HTML contendo painéis, dashboards e/ou modelos com pessoas interessadas, que poderão executá-los em seus navegadores sem nenhuma configuração complicada! 😊

Como o PyScript funciona

PyScript é construído em cima do `Pyodide`, que é uma biblioteca que transforma código Python em WebAssembly. O que traz a próxima pergunta: o que é WebAssembly?

WebAssembly é a tecnologia que possibilita escrever sites em Python. Ela usa uma linguagem de formato de texto `.wat` legível por humanos, que então é convertida para um formato `.wasm` binário que os navegadores podem executar. Graças a isso, podemos escrever código em qualquer linguagem, compilá-lo para o WebAssembly e executá-lo em um navegador da web!

Por enquanto, o PyScript suporta escrever e executar apenas código Python em um navegador. O objetivo para o futuro é que também ofereça suporte para outras linguagens de programação.

É também aqui que entra em jogo uma potencial limitação: atualmente, ao usar o PyScript, podemos usar **apenas** as bibliotecas suportadas pelo `Pyodide`. Você pode encontrar a [lista completa aqui](#).

Seu primeiro código Python no Browser com o PyScript

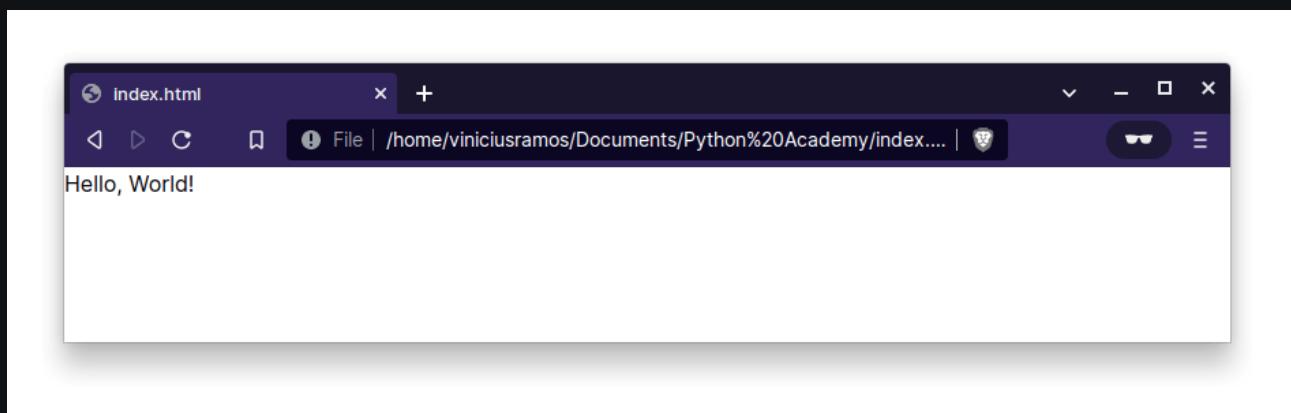
Para começar a brincar com o PyScript, primeiro crie um arquivo `index.html`.

Em seguida, copie o seguinte código, que é o exemplo mais básico - o **Hello World** do PyScript:

```
<html>
  <head>
    <link rel="stylesheet" href="https://pyscript.net/alpha/pyscript.css" />
    <script defer src="https://pyscript.net/alpha/pyscript.js"></script>
  </head>
  <body>
    <py-script> print('Hello, World!') </py-script>
  </body>
</html>
```

Como podemos ver, o código Python está embutido no bloco `<py-script>`.

Abra o arquivo no navegador e verá o seguinte na saída:

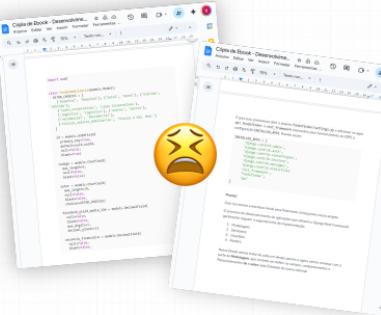


💡 Estou construindo o **DevBook**, uma plataforma que usa IA para criar ebooks técnicos — com código formatado e exportação em PDF. Te convido a conhecêr!

 DevBook

Crie Ebooks técnicos incríveis em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs

Deixe que nossa IA faça o trabalho pesado

 Syntax Highlight  Adicione Banners Promocionais  Edite em Markdown em Tempo Real  Infográficos feitos por IA

TESTE AGORA! PRIMEIRO CAPÍTULO 100% GRÁTIS 

Indo além

No segundo exemplo, faremos um pouco mais!

Usaremos o `numpy` para gerar números provenientes da distribuição Normal Padrão e, em seguida, plotá-los usando `matplotlib`.

Parece mentira, mas é possível!

O código é o seguinte (a explicação vem depois):

```
<html>
  <head>
    <link rel="stylesheet" href="https://pyscript.net/alpha/pyscript.css" />
    <script defer src="https://pyscript.net/alpha/pyscript.js"></script>
    <py-env>
      - numpy
      - matplotlib
    </py-env>
  </head>

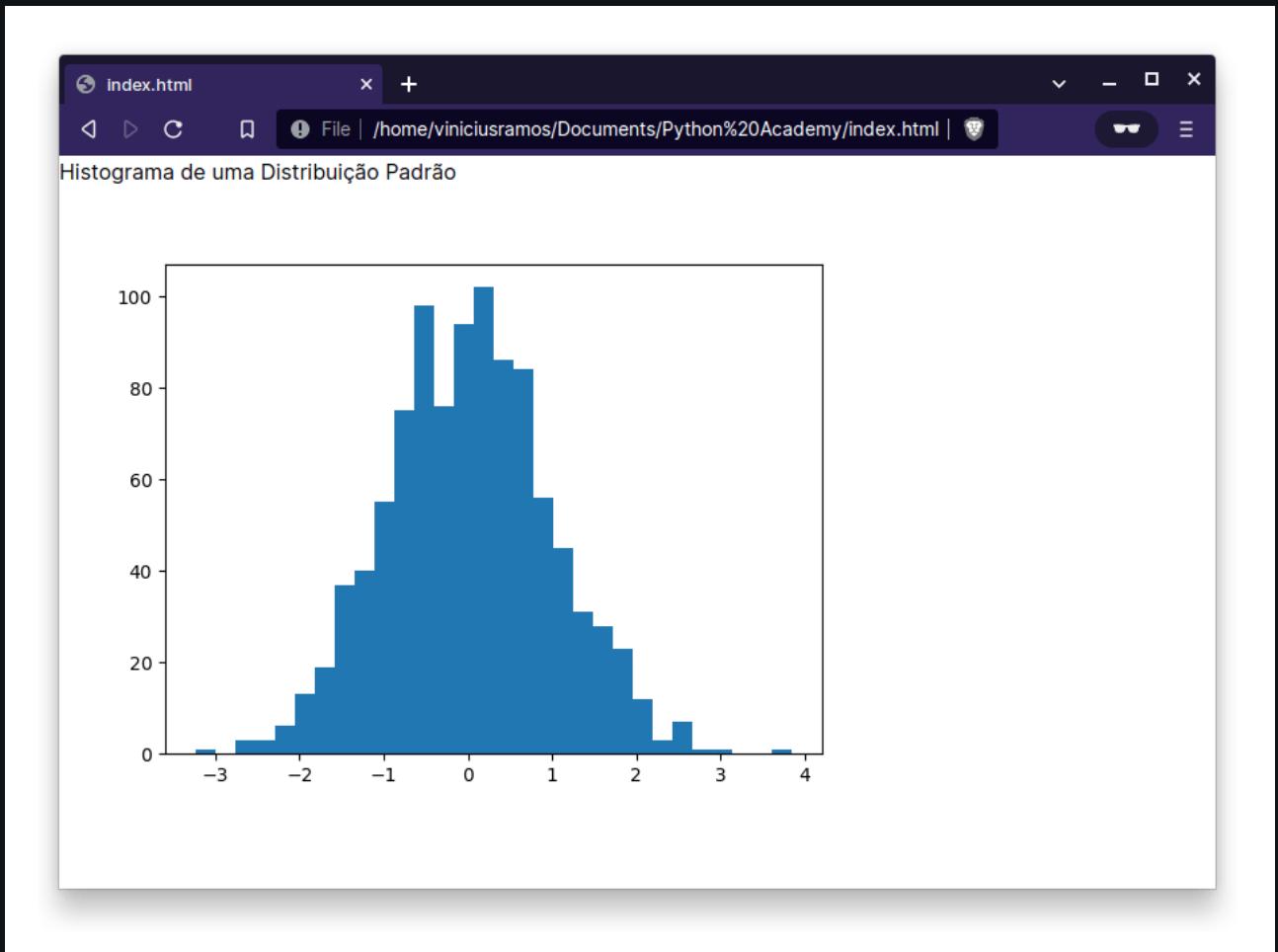
  <body>
    <h1>Histograma de uma Distribuição Padrão</h1>
    <div id="grafico"></div>
    <py-script output="grafico">
import matplotlib.pyplot as plt
import numpy as np

np.random.seed(42)

rv = np.random.standard_normal(1000)

fig, ax = plt.subplots()
ax.hist(rv, bins=30)
fig
    </py-script>
  </body>
</html>
```

E o resultado é esse:



Agora, vamos explicar alguns pontos: - Usamos o bloco `<py-env>` para definir as bibliotecas que vamos utilizar como uma lista, com uma biblioteca por linha - Indicamos que vamos desenhar um gráfico especificando-o no bloco `<py-script output=grafico">`.

Naturalmente, conforme nosso código cresce, é interessante deixar o código HTML mais limpo. Podemos fazer isso, separando o código Python em um arquivo `.py`, da seguinte forma:

```
<html>
  <head>
    <link rel="stylesheet" href="https://pyscript.net/alpha/pyscript.css" />
    <script defer src="https://pyscript.net/alpha/pyscript.js"></script>
  </head>
  <body>
    <py-script src="/arquivo.py"> </py-script>
  </body>
</html>
```

Você pode encontrar diversos exemplos de uso do PyScript [aqui](#) (já em execução no navegador) e [aqui](#) (código no GitHub).

Conclusão

É isso galera, nesse post vimos como podemos executar código Python diretamente no browser com o PyScript, abrindo um leque enorme de possibilidades!

O PyScript parece promissor, mas pode potencialmente abrir muitos novos problemas de segurança.

E neste momento, também podemos ver que mesmo executar alguns scripts simples já gera uma sobrecarga significativa em termos de tempo de execução.

E aí pessoal, o que achou?! Comentem aqui embaixo, quero saber a opinião de vocês!

(Artigo original: [PyScript – unleash the power of Python in your browser](#))

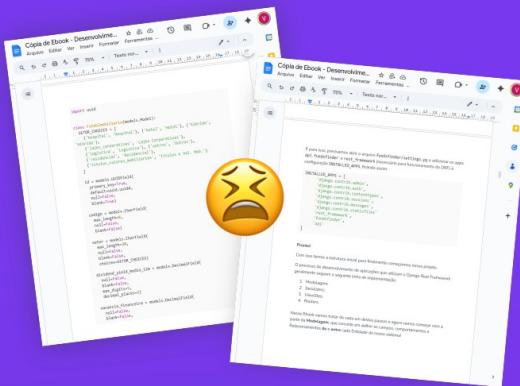
Não se esqueça de conferir!



DevBook

Crie Ebooks técnicos em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs



Syntax Highlight



Adicione Banners Promocionais



• Infográficos feitos para...

Deixe que nossa IA faça o trabalho pesado



 Edite em Markdown em Tempo Real

TESTE AGORA



 PRIMEIRO CAPÍTULO 100% GRÁTIS